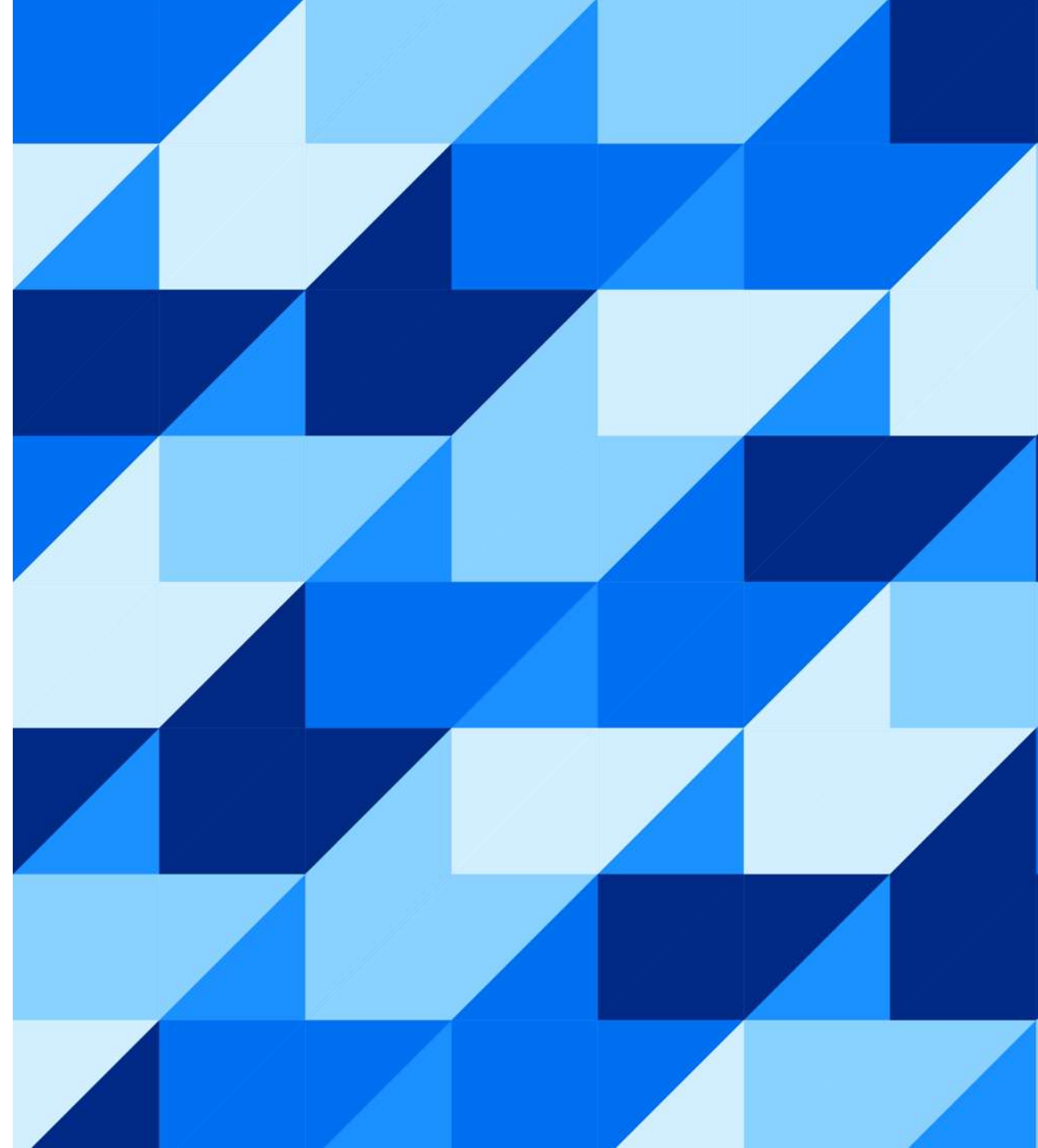


# Frontend Accessibility

## It's not just alt text

Michal Kozubík, SAP  
Development Expert  
May 22, 2024

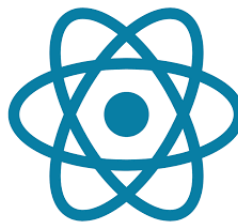


About me

cesnet



Kentico



# What is accessibility?



What is accessibility?

# What is NOT accessibility?



Accessibility is not – ability to access something



Accessibility is not – axe's ability





**Accessibility might be – using axe to control the PC**





# Accessibility should NOT be

- Just „for blind users“
- Optional, if you care about users
  - (and about business)

# What is accessibility?

# Accessibility

- Making the web usable for everyone, regardless of their abilities
- Covers visual, auditory, motor and other disabilities
- Helps everyone – e.g. captions in a noisy room
- It's shared responsibility – UX, dev, testing...
- Is part of EU legislation standards
  - “Directive (EU) 2016/2102 on the accessibility of the websites and mobile applications of public sector bodies, the '[Web Accessibility Directive](#)' or 'WAD', obliges all public sector bodies in the EU to make their online websites and mobile apps accessible.”

# Accessible UX

## Examples



# Accessible UX – don't rely on same screen size

- User must always have a way to access the full information

## Processes

📁 595.5m total cases 🔍 7.6b total events

⬆

All (2)  🔍 [Create](#) ⬆⬆

Name <span>≡</span>	Investigations	Dashboards	Cases	Events	Last edited
mk_tm with a very long name that will be truncated (or maybe not - but most probably yes, as it contains a lot of let...	0	0	877	5k	09/08/2023
Test with a very long name that includes numbers 21302141041501	0	0	0	0	05/23/2024

# Accessible UX – make it easy for the user

- Proper font size
  - Headers, sub headers, paragraphs
- [Right color contrast](#)



**Good contrast**



**Bad contrast**

# Accessible UX – don't rely on user's ability

- “2FA” – 2 factor accessibility
- Quiz – how many colors you see?

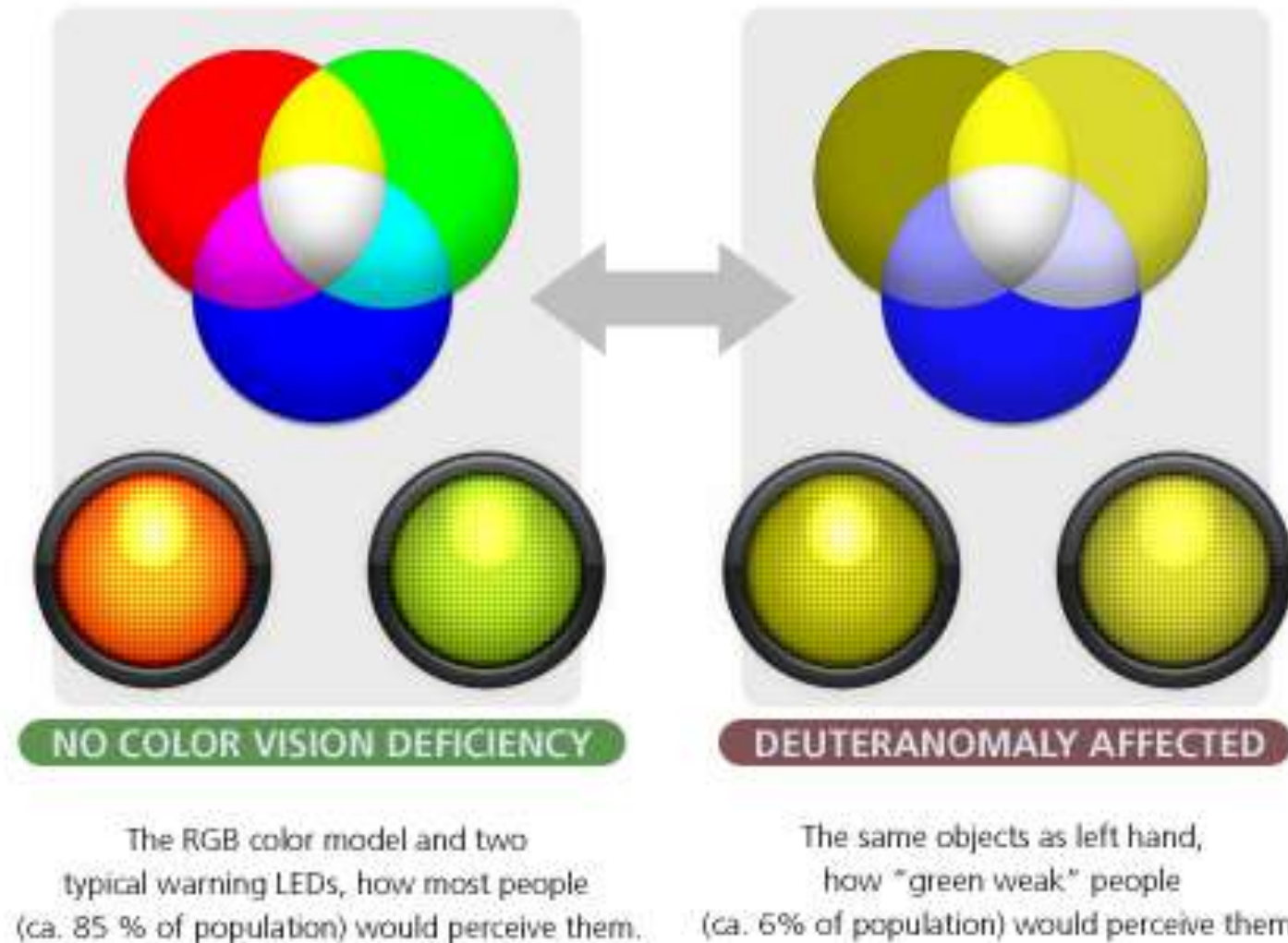


Status: Error



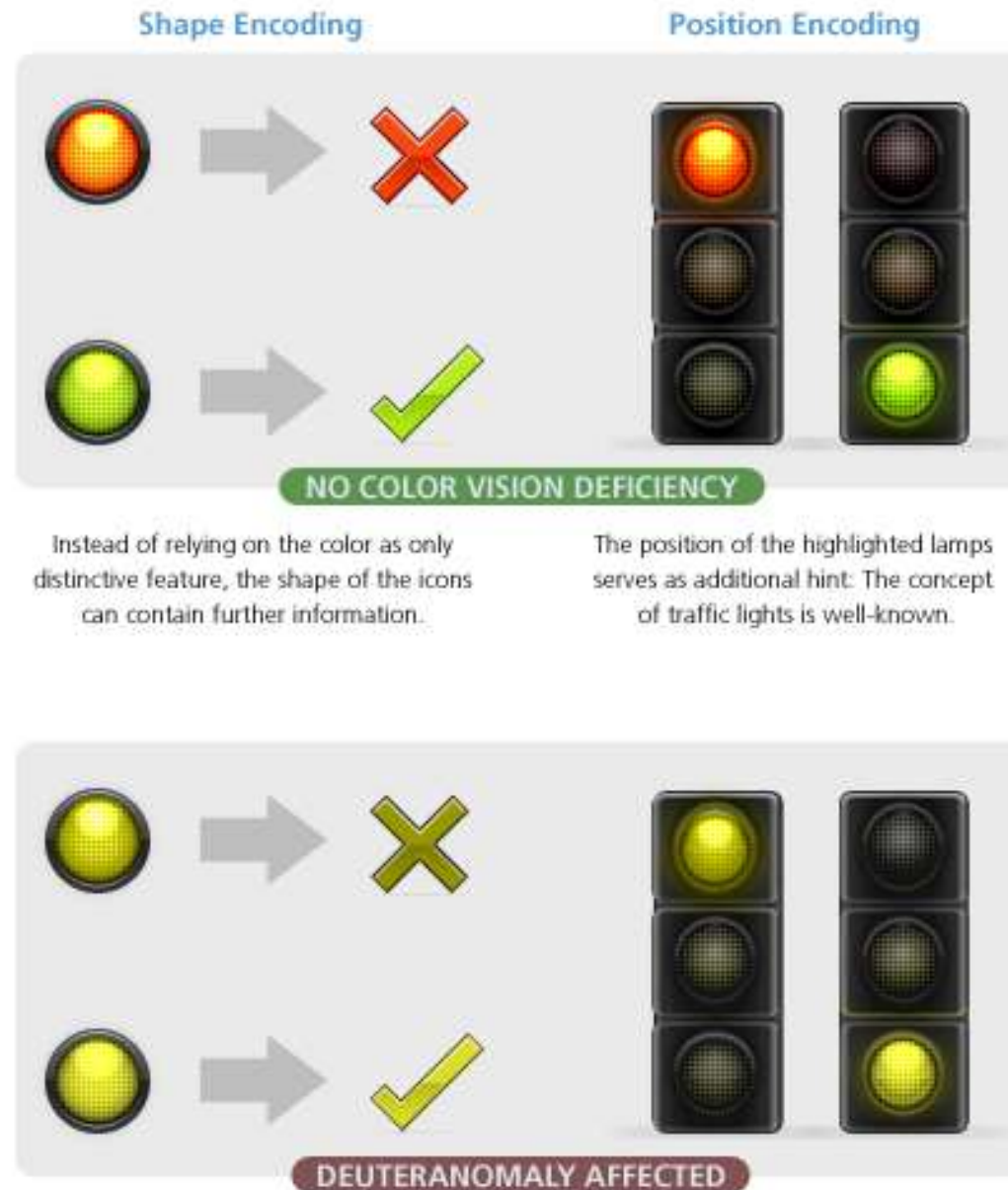
Status: OK

## Accessible UX – don't rely on user's ability





# Accessible UX – don't rely on user's ability



# Accessible UX – don't rely on user's ability

- What blind people see:

# Accessible UX

- because not everyone can see the teeth



# Accessible UI Testing

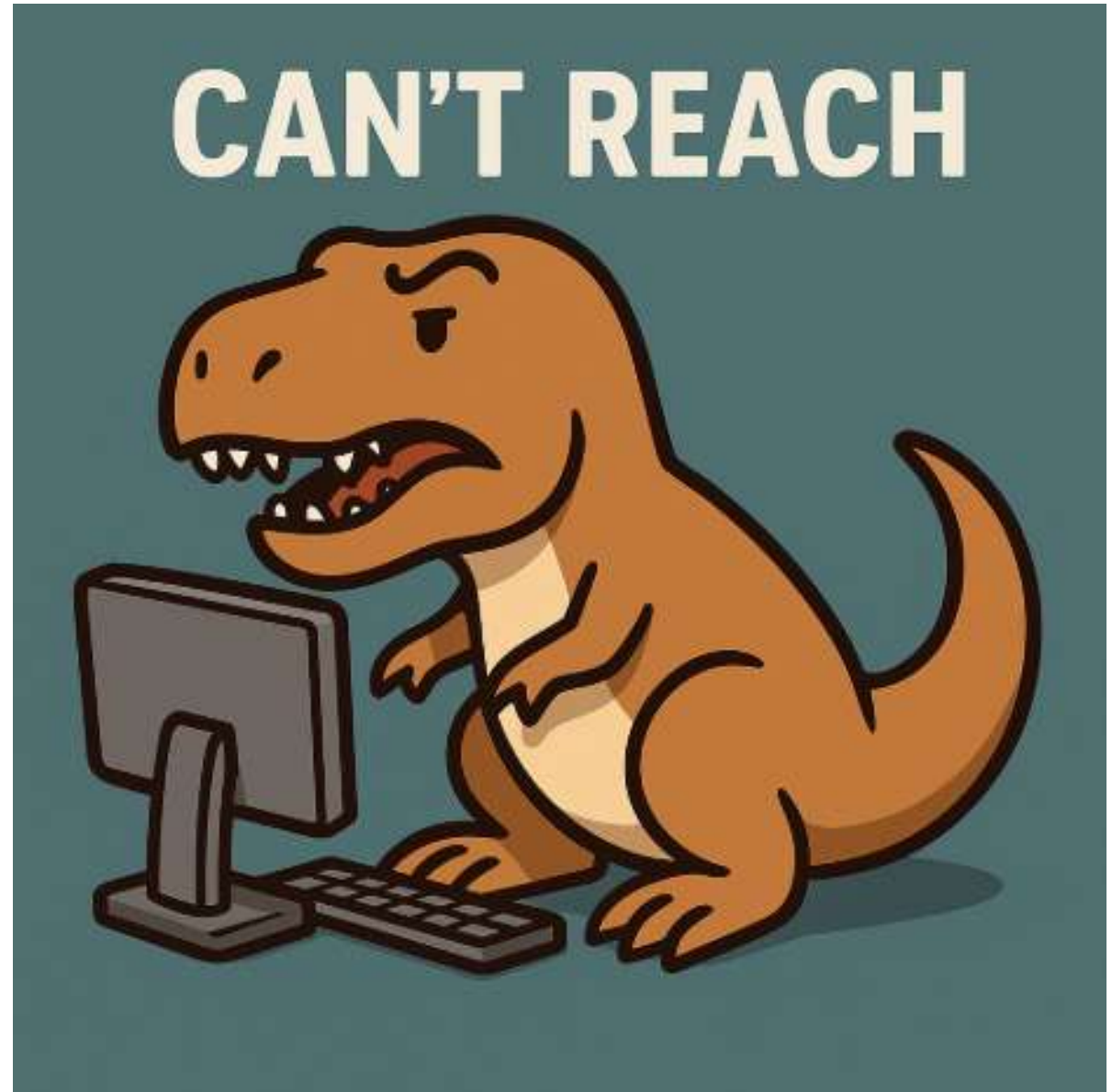


# Accessible UI testing

- Manual testing
  - Keyboard
  - Mouse
  - Screen Readers
    - ChromeVox
    - VoiceOver (MacOS)
- Automated testing
  - Lighthouse
  - axe-core
    - Cypress-axe
    - Jest-axe
    - Playwright-axe
  - Cypress Accessibility

# Accessible UI testing

- T-Rex testing
  - Voice Control
  - Sip-and-puff
  - Eye Tracking
  - Head Tracking



# *HAMILTON* RADIO

*"IT'S  
HAMMERTIME."*



# DEMO TIME

[https://developer.mozilla.org/en-US/docs/Learn\\_web\\_development/Core/Accessibility/HTML](https://developer.mozilla.org/en-US/docs/Learn_web_development/Core/Accessibility/HTML)

Note for Michal: Don't forget the magic you discovered 5 minutes ago

a11y test automation



# a11y test automation – verbose approach

- Pros
  - Explicit testing of every element
  - No need for additional libraries
  - Available in Jest, cypress...
- Cons
  - Explicit testing of every element
  - Easy to miss something
  - A lot of code
  - Can increase test complexity
  - Can increase test run time
  - Single failure = end of test

```
// explicitly check the alt text of an image  
cy.get('img[data-cy="logo"]').should('have.attr', 'alt', 'Cypress Logo')
```

# a11y test automation – verbose approach

## Asserting the accessible name of a button

A similar accessibility-minded locator technique can be used with interactive elements like buttons:

```
// click the "Submit" button located by `cy.contains()`  
cy.contains('button', 'Submit').click()
```

And using the Testing Library `ByRole` locator:

```
// click any element with role `button` and an accessible name of `Submit`  
cy.findByRole('button', { name: 'Submit' }).click()
```

# a11y test automation – axe approach

- Pros
  - Automatic testing of every element
  - Easy integration
  - Open source
  - Covering a lot of standards
  - Configurable
  - Whole report after test end
- Cons
  - Automatic testing of every element
    - But this can be configured
  - Can increase test complexity
  - Can increase test run time
  - Need to call it multiple times to cover different states

```
it("Should have no accessibility violations on good.html", () => {  
  cy.visit("/good.html");  
  cy.injectAxe();  
  cy.checkA11y();  
});
```

# a11y test automation – Cypress Accessibility

- Pros
  - No configuration
  - Using axe under the hood
  - Testing all app states
  - No performance impact
  - Snapshots
  - Run comparison
  - ...
- Cons
  - Only available in Cypress Cloud
  - Paid service
  - A lot of info at once
    - Can be confusing

# a11y test automation – Cypress Accessibility

**Cypress Accessibility** Early Access

92% 1k 774 0 Failed rules 16

Views 11 Rules 88 Filter rules

Rules Elements

- Elements must meet minimum color contrast ratio thresholds Serious 86 721 0
- <li> elements must be contained in a <ul> or <ol> Serious 713 0 0
- Images must have alternate text Critical 269 0 0

Views	Snapshots	Score	Elements	Failed rules
/	257	89%	862 459 0	14
/contacts	69	86%	762 356 0	13
/personal	91	86%	873 482 0	13
/transaction/*	67	87%	19 26 0	7
/transaction/new	94	92%	20 31 0	7
/bankaccounts	24	95%	10 25 0	6
/bankaccounts/new	63	92%	12 12 0	6
/user/settings	62	92%	14 16 0	6

Buttons must have discernible text  
1 element

Ensures buttons have discernible text  
[Learn more](#)

`.MuiIconButton-colorPrimary`

One of these must be fixed:

- Element does not have inner text that is visible to screen readers
- aria-label attribute does not exist or is empty

Elements must only use permitted ARIA attributes 2 elements

Ensures ARIA attributes are not prohibited for an element's role  
[Learn more](#)

`.Cal__Container__root`

Inconclusive result  
aria-label attribute is not well supported on a div with no valid role attribute.

`.Cal__Header__wrapper`

Page must have means to bypass repeated blocks 1 element

Elements must meet minimum color



# a11y test automation – Cypress Accessibility

The screenshot displays the Cypress Accessibility DevTools interface. The left sidebar shows the 'All Views' tab with a 91% compliance score. The selected audit is 'Buttons must have discernible text', which has 2 elements. The first element, '.Mui-disabled', is highlighted with a list of issues: 'One of these must be fixed:' followed by six specific accessibility failures related to screen reader visibility, ARIA attributes, and semantics. The right pane shows a preview of the web application, a 'Transaction Detail' page, with a red box highlighting a button that triggered the error. The browser's address bar shows the URL 'http://localhost:3000/transaction/79DIlqJAqfP'.

← All Views 91%

Buttons must have discernible text 2 elements  
Ensures buttons have discernible text  
[Learn more](#)

❗ .Mui-disabled

❌ One of these must be fixed:

- ❌ Element does not have inner text that is visible to screen readers
- ❌ aria-label attribute does not exist or is empty
- ❌ aria-labelledby attribute does not exist, references elements that do not exist or references elements that are empty
- ❌ Element has no title attribute
- ❌ Element's default semantics were not overridden with role="none" or role="presentation"

❌ .MuiIconButton-colorPrimary

> Images must have alternate text 273 elements

> Form elements must have labels 5 elements

🚨 Serious rules

> ARIA dialog and alertdialog nodes should have an accessible name 1 element

> ARIA hidden element must not be focusable or

Transaction Detail

45x45 paid Ted Parisian

1

Write a comment...

http://localhost:3000/transaction/79DIlqJAqfP

# DEMO TIME

(again)

# Accessible UI Development

# Accessible UI development – HTML First, ARIA Second

- Use semantic HTML whenever possible
- [MDN – HTML: A good basis for accessibility](#)
- Useful attributes
  - aria-label, aria-describedby, role, tabindex, alt
- Keyboard control
- Screen reader support
- KISS is your friend
- Multiple ways to control the application
  - Problem: Drag & Drop

# Key takeaways

- Native HTML first
- Mandatory for some sectors
- MFA – Multi-Factor Accessibility (color + text + sound...)
- Full keyboard navigation support
- Automate a11y tests
- **Accessibility means giving everyone a chance to use your awesome product**



# Useful resources

- UX
  - [Centigrade: Regarding Color Vision Deficiency in the Icon Design Process](#)
- Dev + testing
  - [MDN: Accessibility on the web](#)
  - [Cypress: Accessibility testing](#)
  - [Cypress Accessibility Demo](#)
- Standards + legislative
  - [Web Content Accessibility Guidelines \(WCAG\) 2.1](#)
  - [EN 301 549 \(EU: Web Accessibility Directive\)](#)
  - [Zákon č. 99/2019 Sb. – Zákon o přístupnosti internetových stránek a mobilních aplikací](#)

## PR time

"SAP is not stressful at all."

George, 25 years old



"SAP is amazing,  
I am loving it."

John, 2 years experience



# Thank you.

Contact information:

Michal Kozubík

[Michal.Kozubik@sap.com](mailto:Michal.Kozubik@sap.com)

<https://www.linkedin.com/in/michalkozubik/>

<https://github.com/kozubikmichal/>

